Amendments to the Specification:

Please replace paragraph [0048] beginning at page 9, line 1 with the following amended paragraph:

The systems and techniques described can be used with many different types of documents, including, for example, PORTABLE DOCUMENT FORMAT[[™]] (PDF[[™]]) documents. PDF[[™]] documents are in a format originated by Adobe Systems Incorporated of San Jose, California. A PDF[[™]] document is an example of an electronic document in a platform-independent document format that can define an appearance of the electronic document. This document format can be a platform independent storage format capable of storing many different types of data, including graphics, animation and sound, and the defined appearance can be defined for multiple types of display devices, providing a document originator with control over the look and feel of the document regardless of the final destination device. Using documents in this type of format with the techniques described can result in additional advantages for the resulting systems. For example, the document control system can have an architecture that is not tied to a particular software development platform (e.g., the system can be designed to run on both Java and .NET), and can use platform-independent documents, such as PDF[[™]] documents. Thus, the document control system can readily function across several platforms.

Please replace paragraph [0076] beginning at page 18, line 11 with the following amended paragraph:

When the client 510 needs to take an action with respect to the secured document 545, the client 510 can determine that the document 545 is secured, extract the information identifying the server 520 and the document 545, and send a request 515 to the server 520 corresponding to the action and including the document identifying information. In response to this request, the permissions-broker server 520 can translate the document-permissions information 550 into second document-permissions information 555. The second document-

permissions information 555 can be sent to the client 510 to govern the action with respect to the document 545 at the client 510. The client 510 can be a document viewing application, such as the ADOBE ACROBAT® software provided by Adobe Systems Incorporated of San Jose, California, and the document 545 can be a PDF[(TM)] document.

Please replace paragraph [0113] beginning at page 29, line 28 with the following amended paragraph:

The storage providers need not interpret ACLs. The storage provider can simply store and retrieve ACLs without doing any interpretation of them. When a document is created it can be given an initial ACL, which can be stored in the document and used for offline access control if no other ACL for the document exists locally at the client. The storage interface can provide the methods by which these current and initial ACLs are passed back to the securing or viewing components of the server. In general, there can be two main cases: (1) the content being secured does not have any separate identity outside of the document control system (e.g., the content is an email attachment); (2) the content does have an identity outside of the document control system (e.g., the content is a PDF[(TM)] rendition of a document inside a Documentum® repository). In this latter case, the service provider should be able to dynamically control access to the content in terms of the current rules the repository applies to the object from which the content was derived. Moreover, once an ACL has been saved, it can be modified by the owner, or by a system administrator in the case of a policy.

Please replace paragraph [0148] beginning at page 42, line 10 with the following amended paragraph:

As mentioned above, documents can be secured either at the server or at the client. A document can be converted from one format to another (e.g., from Microsoft Word to PDF[(TM)]) before securing; the document control system can be integrated with a PDF[(TM)] creation service for this purpose. The securer component 960, 990 can be a wrapper around a PDF[(TM)] library that takes a PDF[(TM)] document as input as well as an encryption key and a set of name/value

pairs that represent information to be embedded in the PDF[[™]] document's encrypt dictionary. The securer can encrypt the document with the provided encryption key and embed the specified information in the document. When the securing is performed on the server 900, the securing can be done in a separate process - a pool of such processes can be kept available so that multiple securing requests can be simultaneously satisfied, and the maximum number of such processes can be a configuration option for the server 900. These securing processes can be terminated after some number of successful securing operations, which number can also be a configuration option, or after any unsuccessful securing operation.